UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/675,969	10/02/2003	Hong-Ki Kim	123016-05004702	6294

22429 7590 03/16/2009

LOWE HAUPTMAN HAM & BERNER, LLP
1700 DIAGONAL ROAD
SUITE 300
ALEXANDRIA, VA 22314

EXAMINER	
TERMANINI, SAMIR	

ART UNIT	PAPER NUMBER
2178	

MAIL DATE	DELIVERY MODE
03/16/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/675,969	KIM ET AL.	
	Examiner	Art Unit	
	SAMIR TERMANINI	2178	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *20 November 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-35 and 37-42* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-35 and 37-42* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *02 October 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

BACKGROUND

1. Due to typographical errors, the Office Action having a mail date of 2/18/2009 is hereby withdrawn and this non-final Office Action is issued in its place.

2. This Non-final Office Action is responsive to the following communications: Amendment filed on 11/20/2008.

3. Claim(s) 1-35 and 37-42 are pending. Claim(s) 1, 14, and 23 are independent in form. Claims 39-42 are new.

RESPONSE TO AMENDMENT

4. Arguments concerning the Examiner's Rejections of **Claims 1-3, 8-16, 21-25, 30-35** under 35 U.S.C. 103(a) as being unpatentable over *Bogdan* (US Pat. No. 5,903,265) in view of *Pham et al.* (US Pat. No. 6,301,666 B1) made in the previous Office Action (Mail dated: 11/20/2008) have been fully considered and are persuasive. Those Rejections are hereby withdrawn. However, after searching for the limitations absent from *Rive,* and upon further consideration, a new ground(s) of rejection is made in view of *Pham et al.* (US Pat. No. 6,820,136).

5. The remaining rejections of the 11/20/2008 office action are being maintained as described more fully in detail hereunder.

CLAIM REJECTIONS-35 U.S.C. § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. **Claims 1-3, 8-16, 21-25, 30-35, and 39-42** are rejected under 35 U.S.C. 103(a) as being unpatentable over *Bogdan* (US Pat. No. 5,903,265) in view of *Pham et al.* (US Pat. No. 6,820,136).

As to independent **claim 1**, *Bogdan* teaches a method of controlling an icon appearance (allows a user to customize the size of window elements provided by an operating system, col. 2, lines 59-61) of a display system having a display screen (video display, col. 2, line 26), the method comprising: backing up display properties of the display system ("saving additional system metrics scheme by pressing the "Save Scheme" button 76.," col. 4, lines 27-36; see also e.g. "SetMenuItemInfo() function in that it interrogates information from the MENUITEMINFO structure.," col. 6, lines 16-20; see also SystemParametersInfo(), col. 6, lines 32-40) which are currently set for an original icon appearance (i.e. "CXICON Icon width ...CYICON Icon height...," col. 4, lines 47-67; see also "CXICONSPACING Horizontal icon spacing...CYICONSPACING

Vertical icon spacing" col.4, lines – 9) by generating a first registry subkey in a memory of the display system:

> The SystemParametersInfo() function is provided by the operating system 48 to enable an application to query or set system wide parameters. The system wide parameter to query or set is specified by a parameter that is passed to the function call. Amongst the possible values for this parameter is the SPI.sub.--SETNONCLIENTMETRICS value and the SPI.sub.--GETNONCLIENTMETRICS parameter. These parameter values are specified to either set or retrieve the various sizes of system visual elements that are defined within the operating system 48, as discussed above. (col. 6, lines 32-42)

Examples of first registry subkeys:

> CXICON Icon width
> CYICON Icon height
> CXSIZE Minimize/Maximize icon width
> CYSIZE Minimize/Maximize icon height
> CXICONSPACING Horizontal icon spacing
> CYICONSPACING Vertical icon spacing (col., lines 47-67)

if the display properties are determined to be valid ("...comply with standards that permit its use in the operating system." col. 2, lines 11-12)(emphasis added) and storing the display properties in a corresponding registry; displaying an icon control window on the display screen (dialog box 64, col. 3, lines 36-37), the icon control window including at least one sample icon for a user's preview (icon contained within preview area, preview section 68, col. 3, line 38 ;*see also* e.g. Fig. 5); changing the at least one sample icon's appearance (e.g. icon width, height, horizontal spacing, and vertical spacing VIA elements: "CXICON," "CYICON," "CXICONSPACING," and "CYICONSPACING," respectively, see table spanning cols. 3-4) according to inputs for a new icon appearance

being received from a user through the icon control window ("The user may click the mouse 44 on the upward arrow 84 to increase the element size and click the mouse on the downward arrow 86 to decrease the element size. In addition, the user may put the caret on the value and directly edit the value." col. 4, lines 44-58); and changing the icon appearance of the display system by changing the display properties in accordance with the user inputs ("after the user has finalized the changes and exited the dialog box 64, the bitmaps stored in the bitmap cache 52 (FIG. 3) are re-drawn in response to the user request...," col. 4, lines 52-55) wherein backing up the display properties is performed immediately prior to changing the at least one sample icon's appearance ("The bitmaps are then transferred using the BitBlt() function [f]rom the display drivers to the bitmap cache 52 [step 56]...," col. 3, lines 23-32). See also element 56 in Fig. 4, showing that backing up the display properties is performed immediately prior to changing the at least one sample icon's appearance:

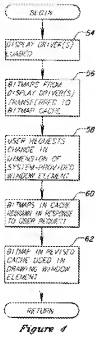


Figure 4 of U.S. Pat. No. 5,903,265

Bogdan teach that the backing up is performed automatically in response to the inputs for a new icon appearance being received from the user through the icon control window ("...Each time that the system metrics are changed, these routines are called to re-draw the bitmaps for the system-provided window elements...," col.4, lines 62-67) .

Bogdan differs from claim 1 in several regards. For example, *Bogdan* does not clearly teach that the backing up of display properties occurring automatically in response to the inputs for a new icon appearance being received from the user through the icon control window and is performed immediately prior to changing the at least one sample icon's appearance.

Pham et al. is cited for teaching the use of a registry for storing properties before changes are made:

> When replicating the source key to the destination key and the latter already exists, the monitoring object will save the original destination key before overwriting it. This action allows the object to be able to restore the destination key with the saved original data when requested.

(col. 5, lines 20-35). Moreover, the display properties to be backed-up are first determined to be valid or not:

> STM1. Check if the monitoring is already in progress. If so, exit with error. If "No", then go to STM2. STM2. Check if the registry key specified by the property RegistryKey is valid. If not, exit with error. If "Yes", go to STM3.

(col. 9, lines 50-55).

It would have been obvious to one ordinary skill in the relevant field at the time the invention was made, to back up of display properties occurring automatically in response to the inputs for a new icon appearance being received from the user through the icon control window immediately prior to changing the at least one sample icon's appearance because *Pham et al.* explain that is it desirable to do so with the same Windows Operating System of *Bogdan*:

> In Windows Operating Systems there is a database designated as the Registry. This Registry database holds all varieties of information from system configurations to performance data to data about the applications available for use in the NT platform. This Registry is kept as a secured file on disk and a typical Registry is illustrated in FIG. 2. The Registry can also be held in memory in addition to the disk file. Each item in the Registry is designated as a Registry Key.

(col. 1, lines 58-67).

They go on to teach:

> The present invention eliminates the need to manually backup Registry keys. By scripting the object, which is the Component Object Model (COM) involved, one can automate the whole process with minimum operator effort and time consumption so that the remote platform will always hold the most up-to-date information. Thus, any applications that keep their settings and other vital information in the local Registry of the local platform can easily make themselves suitable for a fail-over strategy without worrying how the registry data are replicated and maintained in the remote standby platform.

(col. 2, lines 14-25).

Therefore, as can be seen, *inter alia*, from the teachings of *Pham et al.*, the common knowledge, the prior art as a whole, and the nature of the problem itself[1] suggests the use of the registry for storing, securing, and protecting properties from loss.

As to dependent **claim 2**, *Bogdan* further teaches the limitations of claim 1 wherein the received inputs include at least one of an icon size (icon width: "CXICON" and height: "CYICON," see table spanning cols. 3-4; *see also* window element, Fig. 5) vertical icon spacing ("CYICONSPACING," see table spanning cols. 3-4; *see also* window element, Fig. 5), a horizontal icon spacing ("CXICONSPACING," see table spanning cols. 3-4; *see also* window element, Fig. 5), an icon font size ("...changing the

[1] *DyStar Textilfarben GmBH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F.3d 1356, 1361, 80 USPQ2d 1641, 1645 (Fed. Cir. 2006) ("The motivation need not be found in the references sought to be combined, but may be found in any number of sources, including common knowledge, the prior art as a whole, or the nature of the problem itself.")

font size...," col. 6, line 7), and an icon font type (icon under font selection, Fig. 5). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to dependent **claim 3**, *Bogdan* further teaches the limitations of claim 1 wherein the icon control window comprises: an icon size controller providing a plurality of selectable icon sizes for the user to select a desired icon size from the selectable icon sizes ("pre-defined schemes that each specify a single unique set of values [through a] drop down list box 74 that lists the system metrics", col. 4, lines 10-17); a preview region including the at least one sample icon, the sample icon being resized when the desired icon size is selected through the icon size controller ("Examples of window elements that are generated in accordance with the currently selected system metrics scheme are displayed in [preview] section 68."col. 4, lines 20-22)(emphasis added); and an execution controller interfacing with the display system in order to change an icon size of the display system according to the selected icon size ("when an application program wishes to draw a window on the video display, the application program retrieves the bitmaps from the cache and uses the bitmaps to draw the system-provided window elements...," col.2, lines 16-20). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to dependent **claim 8**, *Bogdan* further teachs the limitations of claim 1, wherein the icon control window (e.g. 64, Fig.5) comprises: a plurality of manual input controllers (e.g. plurality of manual input controllers of control window 64, Fig. 5)

manually receiving the inputs from the user ("The user may click the mouse 44 on the upward arrow 84 to increase the element size and click the mouse on the downward arrow 86 to decrease the element size. In addition, the user may put the caret on the value and directly edit the value." col. 4, lines 44-58); a preview region including the at least one sample icon, the sample icon's appearance being changed according to the manually received inputs (section 68, col. 3, lines 36-40); and an execution controller interfacing with the display system for changing the display properties in accordance with the received user inputs ("after the user has finalized the changes and exited the dialog box 64, the bitmaps stored in the bitmap cache 52 (FIG. 3) are re-drawn in response to the user request...," col. 4, lines 52-55; see also "OK" button of 64, Fig. 5). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to dependent **claim 9**, *Bogdan* further teachsthe limitations of claim 8, wherein the user inputs comprises at least one of an icon size (e.g. window element 70 of Fig. 5 is icon width: "CXICON" and height: "CYICON;" *see also* table spanning cols. 3-4), vertical icon spacing (e.g. window element 70 of Fig. 5 is "CYICONSPACING," *see also* table spanning cols. 3-4), horizontal spacing(e.g. window element 70 of Fig. 5 is "CXICONSPACING," *see also* table spanning cols. 3-4), icon font size ("...changing the font size...," col. 6, line 7), and an icon font type (e.g. user input Fonts-> Icon in area 72 of 64 Fig. 5). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to independent **claim 10**, *Bogdan* further teaches the limitations of claim 1, wherein the backing up display properties comprises: determining whether the display properties are valid based on a display properties table of the display system ("...comply with standards that permit its use in the operating system." col. 2, lines 11-12; see also "SystemParametersInfo()," col. 6, lines 32-34). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to dependent **claim 11**, *Bogdan* further teaches the limitations of claim 1, wherein the displaying an icon control window comprises: determining whether the display properties are valid ("...comply with standards that permit its use in the operating system." col. 2,lines 11-12) based on a display properties table of the display system ("SystemParametersInfo()," col. 6, lines 32-34); and displaying the icon control window on the display screen if the display properties are determined to be valid ("...by using a dialog box 64...," col. 3, lines 35-36). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to dependent **claim 12**, *Bogdan* further teaches the limitations of claim 1, wherein the changing the at least one sample icon's appearance comprises: determining whether the inputs for the new icon appearance are received through the icon control window ("The user may click the mouse 44 on the upward arrow 84 to increase the element size and click the mouse on the downward arrow 86 to decrease the element size. In addition, the user may put the caret on the value and directly edit the value."

col. 4, lines 44-58; "...by using a dialog box 64...," col. 3, lines 35-36); and changing at least one of an icon size (icon width: "CXICON" and height: "CYICON," see table spanning cols. 3-4; *see also* window element, Fig. 5), vertical icon spacing ("CYICONSPACING," see table spanning cols. 3-4; *see also* window element, Fig. 5), horizontal icon spacing ("CXICONSPACING," see table spanning cols. 3-4; *see also* window element, Fig. 5), icon font size ("...changing the font size...," col. 6, line 7), and an icon font type (icon under font selection, Fig. 5; also e.g. small cap under 72) of the at least one sample icon according to the new icon appearance if the user inputs are received through the icon control window ("...by using a dialog box 64...," col. 3, lines 35-36). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to dependent **claim 13**, *Bogdan* further teaches the limitations of claim 1, wherein the changing the icon appearance of the display system comprises: determining whether the inputs for the new icon appearance are supported by the display system ("...comply with standards that permit its use in the operating system." col. 2,lines 11-12); and changing at least one of an icon size (icon width: "CXICON" and height: "CYICON," see table spanning cols. 3-4; *see also* window element, Fig. 5), vertical icon spacing ("CYICONSPACING," see table spanning cols. 3-4; *see also* window element, Fig. 5), horizontal icon spacing ("CXICONSPACING," see table spanning cols. 3-4; *see also* window element, Fig. 5), icon font size ("...changing the font size...," col. 6, line 7), and an icon font type (icon under font selection, Fig. 5; also e.g. small cap under 72) of the display system according to the new icon appearance if the user inputs are

supported by the display system ("...and must comply with standards that permit its use in the operating system." col. 2, lines 11-12). Thus, the combination of *Bogdan* and *Pham et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 1 above.

As to independent **claim 14**, this claim differs from claim 1 only in that, this claim is a system claim whereas claim 1 is a method claim. Since *Bogdan* taught the system for carrying out the method of claim 1 (system 36, col. 2, lines 66-67), this claim is rejected for the same reasons set forth in the treatment of claim 1.

As to dependent **claims 15-16, 21-22**, these claims differ from claims 2-3 and 8-9, only in that, these claims are system claims whereas claims 2-3 and 8-9, respectively, are method claims. Since *Bogdan* taught the system for carrying out the method of claims 2-3 and 8-9 (system 36, col. 2, lines 66-67), these claims are rejected for the same reasons set forth in the treatment of claims 1, 2-3 and 8-9 respectively.

As to independent **claim 23**, this claim differs from claim 1 only in that, this claim is a product claim defined by the method of claim 1. Since *Bogdan* taught the product for carrying out the method of claim 1 ("A computer-readable medium having computer-executable instructions for performing, by a computer system having a display and a processor running an operating system and an application program..." see Claim No. 8), this claim is rejected for the same reasons set forth in the treatment of claim 1.

As to dependent **claim 24-25 and 30-35,** these claims differ from claims 2-3 and 8-13 only in that, these claims are product claims defined by the methods of claims 2-3 and 8-13, respectively. Since *Bogdan* taught the product for carrying out the method of claim 1 ("A computer-readable medium having computer-executable instructions for performing, by a computer system having a display and a processor running an operating system and an application program..." see Claim No. 8), these claims are rejected for the same reasons set forth in the treatment of claims 2-3 and 8-13, respectively.

As to **claim 37,** *Bogdan* further teaches the limitations of claim 1 wherein the display properties include one of an icon size, a vertical icon spacing ("CYICONSPACING," see table spanning cols. 3-4; see also window element, Fig. 5, a horizontal icon spacing ("CXICONSPACING," see table spanning cols. 3-4; see also window element, Fig. 5), an icon font size ("...changing the font size...," col. 6, line 7) and an icon font size (icon under font selection, Fig. 5).

As to dependent **claim 38**, *Bogdan* further teaches the limitations of claim 1 wherein the method of claim 37, wherein the change in the sample icon's appearance is performed with respect to the backed-up display properties ("...exited the dialog box 64, the bitmaps stored in the bitmap cache 52 (FIG. 3) are re-drawn in response to the user request (step 60)....," col. 4, lines 50-60).

As to dependent **claim 39**, *Bogdan* further teaches the limitations of claim 1wherein the method of claim 1, further comprising, prior to the changing the icon appearance of the display system ("saving additional system metrics scheme by

pressing the "Save Scheme" button 76.," col. 4, lines 27-36; see also e.g. "SetMenuItemInfo() function in that it interrogates information from the MENUITEMINFO structure.," col. 6, lines 16-20; see also SystemParametersInfo(), col. 6, lines 32-40) temporarily storing the display properties of the display system, which correspond a current icon appearance.

Pham et al. teach storing in a memory location different from where the display properties of the display system, which correspond to the original icon appearance, are backed-up ("remote machine," col. 10, lines 1-5).

It would have been obvious to one ordinary skill in the relevant field at the time the invention was made, to back up of display properties occurring automatically in response to the inputs for a new icon appearance being received from the user through the icon control window immediately prior to changing the at least one sample icon's appearance because *Pham et al.* explain that is it desirable to do so for the same purpose within the same operating system of *Bogdan* (see above).

As to dependent **claim 40**, *Bogdan* further teaches the method of claim 39, further comprising in response to the user's first command, restoring the changed display properties to the temporarily stored display properties (see above).

Pham et al. teach, in response to the user's second command different from the first command, restoring the changed display properties to the backed-up display properties:

> This is a property of the Component which sets or returns the remote backup Registry key name. If the set value is null, a backup

> key identical to the monitored key will be created; otherwise a copy of the monitored key will be created under the specified backup key name

(col. 7, lines 5-12).

It would have been obvious to one ordinary skill in the relevant field at the time the invention was made, to back up of display properties occurring automatically in response to the inputs for a new icon appearance being received from the user through the icon control window immediately prior to changing the at least one sample icon's appearance because *Pham et al.* explain that is it desirable to do so for the same purpose with the same operating system of *Bogdan* (see above).

As to dependent **claim 41**, *Bogdan* further teaches the limitations of claim 1 further comprising: prior to the changing the icon appearance of the display system, temporarily storing the display properties of the display system which correspond a current icon appearance; in response to the user's first command, restoring the changed display properties to the temporarily stored display properties;

Pham et al. teach and in response to the user's second command different from the first command, restoring the changed display properties to the backed-up display properties ("If the set value is null, a backup key identical to the monitored key will be created;" col. 7, lines 5-12).

It would have been obvious to one ordinary skill in the relevant field at the time the invention was made, to restore the back up of display properties because *Pham et*

al. explain that is it desirable to do so for the same purpose with the same operating system of *Bogdan* (see above).

As to dependent **claim 42**, *Pham et al.* further teaches that if the display properties are determined to be invalid, changing the invalid display properties to valid display properties before said generating the first registry subkey:

> However, it is necessary that the remote platform and facilities be in synchronism or have data duplication of the local platform and facilities so that proper operations can occur without using invalid stale data or using data that is no longer correct.

(col. 1, lines 50-60).

It would have been obvious to one ordinary skill in the relevant field at the time the invention was made, to re changing the invalid display properties to valid display properties because *Pham et al.* explain that is it desirable to do so for the same purpose with the same operating system of *Bogdan,* as demonstrated by the quoted teaching.

8. **Claims 4-5, 17-18, and 26-27** are rejected under 35 U.S.C. 103(a) as being unpatentable over *Bogda* and *Morris-Yates et al.* (US PG Pub. 2002/0054144 A1) further in view of *Pham et al.* (US Pat. No. 6,301,666 B1).

As to dependent **claim 4**, *Bogdan teaches* the limitations of claim 3 as discussed above. *Bogdan* does not expressly disclose that the icon size controller comprises a sliding bar with minimum and maximum icon sizes, the user selecting the desired icon size by moving a size indicator within the sliding bar. *Morris-Yates et al.* is cited for

teaching the icon size controller (Fig. 3) comprising a sliding bar with minimum and maximum icon sizes (see Fig. 4 below):

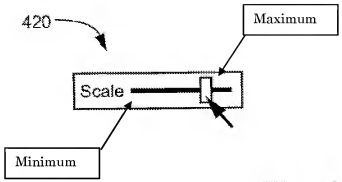


Fig. 4

the user selecting the desired icon size by moving a size indicator (providing active user feedback in a graphic user interface, para. [0003]) within the sliding bar (control 110 being a "scale" adjustment control in a "slider" format, para. [0006]). It would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the sliding bar of *Morris-Yates et al.* in *Bogdan* because *Morris-Yates et al.* is directed to the same problem of using size controllers having sliding bars for scaling graphical elements and expressly suggests the use of the sliding bars for the advantage of providing "...feedback as to the potential results of changing a setting [eliminating the] "change and wait" sequence for the user, which is inconvenient and frustrating." para. [0008]).

As to dependent **claim 5**, *Bogdan* teach the limitations previously discussed with respect to claim 4 above, further comprising the minimum and maximum icon sizes of the sliding bar are selected from a size range supported by the display system

("pre-defined schemes that each specify a single unique set of values [of] system metrics", col. 4, lines 10-17). *Bogdan* does not expressly disclose that the icon size controller comprises a sliding bar with minimum and maximum icon sizes, the user selecting the desired icon size by moving a size indicator within the sliding bar. *Morris-Yates et al.* further teaches the icon size controller (Fig. 3) comprising a sliding bar with minimum and maximum icon sizes (see Fig. 4 above) the user selecting the desired icon size by moving a size indicator (providing active user feedback in a graphic user interface, para. [0003]) within the sliding bar(control 110 being a "scale" adjustment control in a "slider" format, para. [0006]). Thus, the combination of *Bogdan* and *Morris-Yates et al.* meet the claimed limitations for the same reasons set forth in the discussion of claim 4 above.

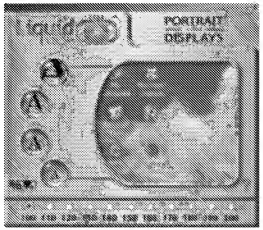
As to dependent **claims 17 and 18** these claims differ from claims 4 and 5 only in that these claims are system claims whereas claims 4 and 5, respectively, are method claims. Since *Bogdan* taught the system for carrying out the method of claims 4 and 5 (system 36, col. 2, lines 66-67), these claims are rejected for the same reasons set forth in the treatment of claims 4 and 5 respectively.

As to dependent **claims 26 and 27,** these claims differ from claims 4 and 5 only in that these claims are product claims defined by the methods of claims 4 and 5, respectively. Since *Bogdan* taught the product for carrying out the method of claim 1 ("A computer-readable medium having computer-executable instructions for performing, by a computer system having a display and a processor running an

operating system and an application program..." see Claim No. 8), these claims are rejected for the same reasons set forth in the treatment of claims 4 and 5, respectively.

9. **Claims 6-7, 19-20, and 28-29** are rejected under 35 U.S.C. 103(a) as being unpatentable over *Bogdan* in view of a publication by Portrait Displays, Inc. titled "Learn How Portait Displays' Liquid View 2.0 Can Bring On-Screen Navigation Into Focus" (hereinafter "*Portrait*").

As to dependent **claim 6**, *Bogdan* teach the limitations of claim 3, discussed above. *Bogdan* does not expressly disclose the icon size controller to comprise a plurality of selectable buttons representing the plurality of selectable icon sizes, the user selecting the desired icon size by selecting one of the selectable buttons. *Portrait* is cited for teaching the icon size controller comprising a plurality of selectable buttons representing the plurality of selectable icon sizes, the user selecting the desired icon size by selecting one of the selectable buttons from 100 to 300 (Eleven predefined settings, col. 1 within text box; *See also* Fig. *Liquid View User Interface*, pp.1 below):



Selectable icon sizes
via selectable buttons

It would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the selectable buttons of *Portrait* with *Bogdan* because *Portrait* is: (1) directed to precisely the same problem of controlling the display of a system having a display screen ("...which gives users an immediate way to increase the size of fonts, icons, and menus...," col. 1, second to last para.)(emphasis added); (2) is in the same field of endeavor of "...letting users quickly scale up their menus and icons..." (col. 1, last paragraph); and (3) *Portrait* expressly suggests "Liquid View 2.0 makes relationships between various user-definable elements simple and easy to change from one location."

The combination of *Bogdan* in view of a publication by Portrait Displays, Inc. differs from claim 6 in that it does not clearly teach that the backing up of display properties occurring automatically in response to the inputs for a new icon appearance being received from the user through the icon control window and is performed immediately prior to changing the at least one sample icon's appearance.

Pham et al. is cited for teaching the use of a registry for storing properties before changes are made:

> When replicating the source key to the destination key and the latter already exists, the monitoring object will save the original destination key before overwriting it. This action allows the object to be able to restore the destination key with the saved original data when requested.

(col. 5, lines 20-35). Moreover, the display properties to be backed-up are first determined to be valid or not:

> STM1. Check if the monitoring is already in progress. If so, exit with error. If "No", then go to STM2. STM2. Check if the registry key specified by the property RegistryKey is valid. If not, exit with error. If "Yes", go to STM3.

(col. 9, lines 50-55).

It would have been obvious to one ordinary skill in the relevant field at the time the invention was made, to back up of display properties occurring automatically in response to the inputs for a new icon appearance being received from the user through the icon control window immediately prior to changing the at least one sample icon's appearance because *Pham et al.* explain that is it desirable to do so with the same Windows Operating System of *Bogdan*:

> In Windows Operating Systems there is a database designated as the Registry. This Registry database holds all varieties of information from system configurations to performance data to data about the applications available for use in the NT platform. This Registry is kept as a secured file on disk and a typical Registry is illustrated in FIG. 2. The Registry can also be held in memory in addition to the disk file. Each item in the Registry is designated as a Registry Key.

(col. 1, lines 58-67).

They go on to teach:

> The present invention eliminates the need to manually backup Registry keys. By scripting the object, which is the Component Object Model (COM) involved, one can automate the whole process with minimum operator effort and time consumption so that the remote platform will always hold the most up-to-date information. Thus, any applications that keep their settings and other vital information in the local Registry of the local platform can easily make themselves suitable for a fail-over strategy without worrying how the registry data are replicated and maintained in the remote standby platform.

(col. 2, lines 14-25).

Therefore, as can be seen, *inter alia*, from the teachings of *Pham et al.*, the common knowledge, the prior art as a whole, and the nature of the problem itself[2] the use of the registry for storing, securing, and protecting properties from loss was suggested.

As to dependent **claim 7**, *Bogdan* teach the limitations of claim 6 as discussed above. *Bogdan* does not expressly disclose that the plurality of selectable buttons include toggle buttons. *Portrait* is cited for teaching the plurality of selectable buttons include toggle buttons (see above, Fig. Liquid View User Interface, where each of the eleven buttons are toggle buttons). Therefore it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the toggle buttons of *Portrait* with *Bogdan* for the reasons set forth above.

As to dependent **claims 19 and 20** these claims differ from claims 6 and 7 only in that, these claims are system claims whereas claims 6 and 7, respectively, are method claims. Since *Bogdan* taught the system for carrying out the method of claims 6 and 7 (system 36, col. 2, lines 66-67), these claims are rejected for the same reasons set forth in the treatment of claims 6 and 7 respectively.

As to dependent **claims 28 and 29,** these claims differ from claims 6 and 7 only in that these claims are product claims defined by the methods of claims 6 and 7.

---

[2] *DyStar Textilfarben GmBH & Co. Deutschland KG v. C.H. Patrick Co.*, 464 F.3d 1356, 1361, 80 USPQ2d 1641, 1645 (Fed. Cir. 2006) ("The motivation need not be found in the references sought to be combined, but may be found in any number of sources, including common knowledge, the prior art as a whole, or the nature of the problem itself.")

respectively. Since *Bogdan* taught the product for carrying out the method of claim 1 ("A computer-readable medium having computer-executable instructions for performing, by a computer system having a display and a processor running an operating system and an application program..." see Claim No. 8), these claims are rejected for the same reasons set forth in the treatment of claims 6 and 7, respectively.

RESPONSE TO ARGUMENTS

10. Arguments concerning the Examiner's rejections of Claims 1-3, 8-16, 21-25, 30-35 under 35 U.S.C. 103(a) as being unpatentable over *Bogdan* (US Pat. No. 5,903,265) in view of *Pham et al.* (US Pat. No. 6,820,136) within in the previous Office Action (Mail dated: 11/20/2008) have been fully considered but are moot in view of the new ground(s) of rejection, addressed, above.

The remaining substance of applicant's arguments are addressed by the new ground(s) of rejection is made in his office action.

CONCLUSION

11. All prior art made of record in this Office Action or as cited on form PTO-892 notwithstanding being relied upon, is considered pertinent to applicant's disclosure. Therefore, Applicant is required under 37 CFR §1.111(c) to consider these references fully when responding to this Office Action.

12. Any inquiry concerning this communication or earlier communications from the examiner should be directed to SAMIR TERMANINI whose telephone

number is (571) 270-1047. The examiner can normally be reached on Monday through Friday (Excluding Alternating Fridays) 8:00 A.M. to 4:30 P.M..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Stephen S. Hong can be reached on (571) 272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Samir Termanini/
Examiner, Art Unit 2178

/Stephen S. Hong/
Supervisory Patent Examiner,
Art Unit 2178